# Package 'gDefrag'

May 23, 2025

**Type** Package

**Title** Graph-Based Landscape De-Fragmentation

**Version** 1.4

**Date** 2025-05-23

**Depends** sp, R (>= 3.0)

**Imports** igraph, methods, terra

**Author** Frederico Mestre, Fernando Ascensao, A. Marcia Barbosa

**Maintainer** Frederico Mestre <mestre.frederico@gmail.com>

**Description** Provides a set of tools to help the de-fragmentation process. It works by prioritizing the different sections of linear infrastructures (e.g. roads, power-lines) to increase the available amount of a given resource.

**License** GPL (>= 2)

**RoxygenNote** 7.3.2

## Contents



---

`gDefrag-package` *Graph-Based Landscape De-Fragmentation*

---

### Description

Provides a set of tools to help the de-fragmentation process. It works by prioritizing the different sections of linear infrastructures (e.g. roads, power-lines) to increase the available amount of a given resource.

## Details

Our approach is based on graph-based tools, and relies only on the spatial configuration of the infrastructures and the layer of the resource to be evaluated (e.g. habitat area or quality). The package interprets the landscape as a graph-like structure, taking advantage of the structural simplicity of this approach. The main assumption is that connecting a larger amount of a given resource is benefit for the species persistence, following the concept of 'accessible habitat' by Eigenbrod et al. (2008). It then prioritizes the sections that have higher impact in overall connectivity according to three methods, including the Integral Index of Connectivity (IIC) developed by Pascual-Hortal and Saura (2006). It can be used to assist in the de-fragmentation process caused by any linear infrastructure type, including roads, railways, power lines, pipelines, water channels, and fences. Hereafter we will refer to these simply as 'roads'.

## Author(s)

Frederico Mestre, Fernando Ascensao, A. Marcia Barbosa

Maintainer: Frederico Mestre <mestre.frederico@gmail.com>

## References

Eigenbrod, F., Hecnar, S.J., Fahrig, L. (2008). Accessible habitat: an improved measure of the effects of habitat loss and roads on wildlife populations. Landscape Ecology, 23(2), 159-168.

Grilo, C., Ascensao, F., Santos-Reis, M., Bissonette, J.A. (2011). Do well-connected landscapes promote road-related mortality?. European Journal of Wildlife Research, 57(4), 707-716.

Pascual-Hortal, L., Saura, S. (2006). Comparison and development of new graph-based landscape connectivity indices: towards the priorization of habitat patches and corridors for conservation. Landscape Ecology, 21(7), 959-967.

Santos, S.M., Marques, J.T., Lourenco, A., Medinas, D., Barbosa, A.M., Beja, P., Mira, A. (2015). Sampling effects on the identification of roadkill hotspots: implications for survey design. Journal of Environmental Management, 162, 87-95.

---

`edge.creation` *Creating graph edges*

---

## Description

Generates the graph edges, connecting nodes that are the landscape polygons resulting from the landscape clip by the road network.

## Usage

```
edge.creation(nodes, land_polyg, min_length = 0,
min_pol_area = 0, plot = TRUE, shape = FALSE,
shape_name_edges = "shape_edges")
```

## Arguments

| | |
|---|---|
| nodes | Output of the function node.creation. |
| land_polyg | SpatialPolygonsDataFrame loaded with readOGR function of the package rgdal (polygon shapefile). |
| min_length | Minimum length of the road segment to consider creating an edge representing it. |
| min_pol_area | Minimum area for a given polygon to be considered a node. |
| plot | Plot after running? (TRUE/FALSE). |
| shape | Create a shapefile with the edges (TRUE/FALSE). |
| shape_name_edges | Name of the edges shapefile to be created (without prioritization). |
| overwrite | Should the shapefile file be overwritten? |

## Details

If shape = TRUE, this function produces one shapefile with edges without prioritization metrics.

## Value

The function returns a SpatVector with the edges. The data table of this data frame has the following fields:

- node_A - Node A Id.
- node_B - Node B Id.
- distance - Euclidean distance between nodes A and B (edge length).
- x_node_A - Node A longitude.
- y_node_A - Node A latitude.
- x_node_B - Node B longitude.
- y_node_A - Node A latitude.
- raster_value_A - Average value of the underlying raster in the polygon represented by node A.
- raster_value_B - Average value of the underlying raster in the polygon represented by node B.
- road_ID - Id of the road represented by the edge.
- road_length - Length of the road represented by the edge.

## Author(s)

Frederico Mestre, Fernando Ascensao and A. Marcia Barbosa

## See Also

node.creation

## Examples

```
data(road_P)

#Obtaining nodes
out1 <- node.creation(land_polyg = road_P,
                      value_col = "frst_sm",
                      scale_nodes = 10,
                      cex_labels = 1,
                      shape = TRUE,
                      shape_name_nodes = "shape_nodes_file",
                      overwrite = TRUE)

#Obtaining edges
out2 <- edge.creation(nodes = out1,
                      land_polyg = road_P,
                      min_length = 0,
                      min_pol_area = 0,
                      shape_name_edges = "shape_edges_file",
                      shape = TRUE,
                      overwrite = TRUE)
```

---

# gDefrag.full *Wrapper function running the full process*

---

## Description

This function runs the full process of creating the nodes, creating the edges and establishing priority edges to be used to enhance connectivity in the landscape for the focal species.

## Usage

```
gDefrag.full(land_polyg, method, value_col = NULL,
min_length = 0, min_pol_area = 0, shape = FALSE,
shape_name_nodes = "shape_all_nodes", shape_name_edges = "shape_edges",
shape_name_out = "priorities_shape",
shape_name_nodes_edges = "nodes_with_edges",
scale_nodes = 10, col_nodes = "deepskyblue4", col_labels = "white",
cex_labels = 1, main = "Graph")
```

## Arguments

| | |
|---|---|
| land_polyg | SpatialPolygonsDataFrame loaded with readOGR function of the package rgdal (polygon shapefile). To be internally passed to as node.creation and as edge.creation. |
| method | Method to be used in the prioritization of the connectivity corridors. Detailed description in the 'details' section. To be internally passed to as prioritize. |
| value_col | Identification of the column in the 'land_polyg' object. To be internally passed to as prioritize. |
| min_length | Minimum length of the road segment to consider creating an edge representing it. To be internally passed to as edge.creation. |

| | |
|---|---|
| min_pol_area | Minimum area of the polygon to consider creating a node representing it. To be internally passed to as node.creation. |
| shape | Produce node and edges shapefiles as output? (TRUE/FALSE) To be internally passed to as node.creation and as edge.creation. |
| shape_name_nodes | Name of the nodes shapefile to be created (all nodes). |
| shape_name_edges | Name of the edges shapefile to be created (without prioritization). |
| shape_name_out | Name of the edges prioritization output shapefile to be created. |
| scale_nodes | Node size scaling factor for correct viewing in the plot. To be internally passed to as plotgraph. |
| col_nodes | Node colour. To be internally passed to as plotgraph. |
| col_labels | Labels colour. To be internally passed to as plotgraph. |
| cex_labels | Text labels font size. To be internally passed to as plotgraph. |
| main | Main plot title. To be internally passed to as plotgraph. |
| overwrite | Should the shapefile file be overwritten? |

## Details

If shape = TRUE, this function produces four shapefiles: all nodes, nodes with area greater than 'min_pol_area', edges without prioritization metrics, edges with prioritization metrics. The prioritization of the connectivity corridors can be made with one of four methods (argument 'method'):

- 'value' - Give priority to edges connecting nodes with higher attribute value (e.g. suitability for the focal species);
- 'traffic' - Give priority to edges representing roads with more traffic;
- 'IIC' - Give priority to edges with higher overall impact in the connectivity. This was implemented by using the Integral Index of Connectivity (IIC) developed by Pascual-Hortal and Saura (2006). The connectivity relevance of each edge was evaluated usinf the approach described in Pascual-Hortal and Saura (2006): dIIC=((I1-I2)/I1)*100, where I1 is the IIC before edge removal and I2 is the IIC after.
- 'between' - Give priority to edges with higher betweenness using the function edge.beetweenness.estimate from the igraph package.
- 'AWM' - Give priority to edges connecting nodes with more suitable area.

## Value

This function produces a SpatVector with the priorization results.

## Author(s)

Frederico Mestre, Fernando Ascensao and A. Marcia Barbosa

## See Also

node.creation, edge.creation, plotgraph, prioritize

## Examples

```
data(road_P)

out1 <- gDefrag.full(land_polyg = road_P, method = "value",
                     value_col = "frst_sm", main = "value-based graph",
                     shape_name_nodes = "fullrun_shape_all_nodes0",
                     shape_name_edges = "fullrun_shape_edges0",
                     shape_name_out = "fullrun_priorities_shape0",
                     shape = TRUE,
                     overwrite = TRUE)
```

---

# node.creation *Creating graph nodes*

---

## Description

Generates the nodes that are the landscape polygons resulting from the landscape clip by infrastructure network.

## Usage

```
node.creation(land_polyg, value_col, plot = TRUE,
scale_nodes = 1, col_nodes = "deepskyblue4", cex_labels = 1,
shape = FALSE, shape_name_nodes = "shape_nodes")
```

## Arguments

| | |
|---|---|
| land_polyg | SpatialPolygonsDataFrame loaded with readOGR function of the package rgdal (polygon shapefile). |
| value_col | Which column in the SpatialPolygonsDataFrame has the relevant information on the underlying raster. This information must be retrieved previously using a GIS, if the user wishes to use the method 'value' for prioritizing edges. |
| plot | Plot the output (TRUE/FALSE). |
| scale_nodes | Scaling factor to represent the nodes graphically. |
| col_nodes | Nodes colour. |
| cex_labels | Font size of the labels. |
| shape | Create a shapefile with the edges (TRUE/FALSE). |
| shape_name_nodes | Name of the nodes shapefile to be created (all nodes). |
| overwrite | Should the shapefile file be overwritten? |

## Details

If shape = TRUE, this function produces one shapefile with all the nodes.

## Value

The function returns a SpatVector with the nodes. The data table of this data frame has the following fields:

- X - Node latitude.
- Y - Node longitude.
- pol_value - Underlying raster value summed for each polygon.
- pol_area - Polygon area.

## Author(s)

Frederico Mestre, Fernando Ascensao and A. Marcia Barbosa

## See Also

edge.creation

## Examples

```
data(road_P)

out1 <- node.creation(land_polyg = road_P,
                      value_col = "frst_sm",
                      scale_nodes = 10,
                      cex_labels = 1,
                      shape = TRUE,
                      shape_name_nodes = "shape_nodes_file",
                      overwrite = TRUE)
```

---

# plotgraph *Plotting the spatial graph*

---

## Description

Function to plot the spatial graph

## Usage

```
plotgraph(nodes, edges, land_polyg, scale_nodes = 10, col_nodes = "darkblue",
col_labels = "darkblue", cex_labels = 1, main = "Graph")
```

## Arguments

| | |
|---|---|
| nodes | Output of node.creation. |
| edges | Output of edge.creation. |
| land_polyg | SpatialPolygonsDataFrame with the polygons defined by roads and the study site limits. |
| scale_nodes | Node size scaling factor for correct viewing in the plot. |
| col_nodes | Node colour. |

| | |
|---|---|
| col_labels | Labels colour. |
| cex_labels | Text labels font size. |
| main | Main plot title. |

## Details

Produces a plot of the study site, nodes and edges with the prioritization order.

## Value

This function plots the spatial graph.

## Author(s)

Frederico Mestre, Fernando Ascensao and A. Marcia Barbosa

## See Also

node.creation, edge.creation, prioritize, road_P

## Examples

```
data(road_P)

#Obtaining nodes
out1 <- node.creation(land_polyg = road_P, value_col = "forest_sum",
scale_nodes = 10, col_nodes = "pink", cex_labels = 1)

#Obtaining edges
out2 <- edge.creation(nodes = out1, land_polyg = road_P,
min_length = 0, min_pol_area = 0)

#Prioritize
out3 <- prioritize(nodes = out1, edges = out2, method = "value")

#Plotting results
plotgraph(nodes = out1, edges = out3, land_polyg = road_P, main = "Habitat value")
```

---

# prioritize *Prioritizing connectivity corridors*

---

## Description

Function to stablish a priority on the edges (connections) to be created in order to de-fragment a landscape.

## Usage

```
prioritize(nodes, edges, method, shape=FALSE, shape_name_out = "priorities_shape",
shape_name_nodes_edges = "nodes_with_edges")
```

## Arguments

| | |
|---|---|
| nodes | Output of node.creation. |
| edges | Output of edge.creation. |
| method | Method to be used in the prioritization of the connectivity corridors. Detailed description in the 'details' section. |
| re_scale | TRUE/FALSE. Should the output be re-scaled between 0 and 100? |
| shape | Create a shapefiles? (TRUE/FALSE). |
| shape_name_out | Name of the shapefile to be created. |
| overwrite | Should the shapefile file be overwritten? |

## Details

If shape = TRUE, this function produces two shapefiles: nodes with area greater than 'min_pol_area' (in edge.creation), edges with prioritization metrics. The prioritization of the connectivity corridors can be made with one of these methods (argument 'method'):

- 'value' - Give priority to edges connecting nodes with higher attribute value (e.g. suitability for the focal species);
- 'IIC' - Give priority to edges with higher overall impact in the connectivity. This was implemented by using the Integral Index of Connectivity (IIC) developed by Pascual-Hortal and Saura (2006). The connectivity relevance of each edge was evaluated usinf the approach described in Pascual-Hortal and Saura (2006): dIIC=((I1-I2)/I1)*100, where I1 is the IIC before edge removal and I2 is the IIC after.
- 'between' - Give priority to edges with higher betweenness using the function edge.beetweenness.estimate from the igraph package.
- 'AWM' - Area Weighted Metric - The user should provide the value of the variable to be weighted by area (e.g. habitat area) in the 'value_col' argument of the function node.creation. The expression for the Area Weighted Metric is: AWM = (value_A*area_B)+(value_B*area_A), where: 'value_A' and 'area_B' are the node A argument and the habitat in polygon B, respectively and 'value_B' and 'area_A' are the node B argument and the habitat in polygon A, respectively.

## Value

The output is a SpatVector with an additional field, 'priorization'. This field provides the desired priorization value.

## Note

The computation of IIC is made resorting to the node attribute in place of the 'habitat patch areas' (ai and aj) as originally devised by Pascual-Hortal & Saura (2006).

## Author(s)

Frederico Mestre, Fernando Ascensao and A. Marcia Barbosa

## References

Pascual-Hortal, L., Saura, S. (2006). Comparison and development of new graph-based landscape connectivity indices: towards the priorization of habitat patches and corridors for conservation. Landscape Ecology, 21(7), 959-967.

## See Also

node.creation, edge.creation

## Examples

```
data(road_P)

####### To the value, betweenness and connectivity metrics #######
#Obtaining nodes
out1 <- node.creation(land_polyg = road_P,
                      value_col = "frst_sm",
                      scale_nodes = 10,
                      cex_labels = 1,
                      shape = TRUE,
                      shape_name_nodes = "shape_nodes_file",
                      overwrite = TRUE)

#Obtaining edges
out2 <- edge.creation(nodes = out1,
                      land_polyg = road_P,
                      min_length = 0,
                      min_pol_area = 0,
                      shape_name_edges = "shape_edges_file",
                      shape = TRUE,
                      overwrite = TRUE)

#Prioritize
out3 <- prioritize(nodes = out1,
                   edges = out2,
                   method = "value",
                   shape=TRUE,
                   shape_name_out = "priorities_shape2",
                   overwrite = TRUE)
```

---

# road_P *Road polygons dataset with habitat area*

---

## Description

Sample dataset. SpatialPolygonsDataFrame with the polygons defined by roads and the study site limits. The field characterizing the raster is the habitat area in the polygon.

## Usage

```
data("road_P")
```

## Format

An object of the class 'SpatVector' [package "terra"] with eleven polygons. The argument table should have a column with the values of an underlying raster (e.g. averaged or summed).

## Details

This dataset has the spatial information on the polygons in the sample study site. The polygons are defined by the roads and the limits of the sample study site.

## See Also

node.creation, edge.creation, prioritize

## Examples

```
data(road_P)
str(road_P)
```

# Index